Entropie de Shannon et codage

Exercice 1 Entropie de Shannon et codage

Soit $A = \{\alpha_1, ..., \alpha_D\}$ un alphabet et $M = \{x_1, ..., x_k\}$ un ensemble d'objets appelés messages. Un codage est une application de M dans A^* qui à un message x_i associe son code c_i de longueur l_i.

Un codage est dit ***préfixe*** s'il ne contient pas deux mots dont l'un est préfixe de l'autre.

▷ 1. *(Inégalité de Kraft)* Montrer que si le codage h est préfixe alors

$$\sum_{i=1}^{k} D^{-l_i} \leqslant 1$$

▷ 2. Supposons l'inégalité précédente satisfaite. Montrer que pour tout alphabet A de D lettres et tout ensemble de k messages il existe un codage h préfixe et dont les codes sont de longueurs (l_i).

▷ 3. *(Borne de Shannon sur la longueur moyenne d'un code préfixe)* Supposons maintenant que les messages proviennent d'une source aléatoire, le message x_i ayant la probabilité p_i. La longueur moyenne de codage est alors définie par :

$$L(h) = \sum_{i=1}^{k} p_i l_i$$

Soit L_{inf} la plus petite de ces longueurs pour un code préfixe. Nous allons estimer L_{inf}.
Shannon définit l'entropie d'un événement comme la quantité de surprise $-\log p_i$ qu'aurait un observateur lorsqu'il découvre la réalisation de cet événement. Plus cet événement est improbable plus l'observateur sera surpris. Si on fait la moyenne sur tous les événements possibles on obtiendra l'entropie du système :

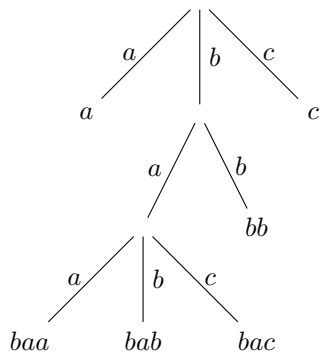
$$H_D(p) = -\sum_{i=1}^{k} p_i \log_D p_i$$

C'est la quantité d'information relative à la distribution $p = (p_1, ..., p_k)$.
Montrer que

$$H \leqslant L_{inf} \leqslant H + 1$$

Solution

▷ 1.



On peut représenter un codage préfixe sur un alphabet à D lettres par un arbre D-aire (pas nécessairement complet). L'ensemble de mots est l'ensemble des étiquettes d'un chemin de la racine à une feuille. À chaque feuille correspond donc un mot du code. Les profondeurs des feuilles sont les longueurs des codes.

On montre par induction sur un tel arbre que l'inégalité de Kraft est vérifiée :

- le cas de base (une feuille) est trivial.
- Considérons un arbre fini où l'inégalité est vérifiée pour les fils de la racine : $\forall j, \sum_{i=1}^{k_j} D^{-l_{i,j}} \leqslant 1$.

  Dans l'arbre de départ, les profondeurs sont augmentées de 1 : $\sum_{i,j}^{k} D^{-l_{i,j}-1} \leqslant \frac{1}{D} \sum_{j=1}^{D} 1 = 1$.

▷ 2. Pour établir la réciproque, soit $L := \max_i \ell_i$, et considérons un arbre D-aire complet de profondeur L. On prend les ℓ_i un par et pour chacun,

- on choisit arbitrairement un nœud à profondeur ℓ_i et le mot étiquetant le chemin de la racine à ce nœud
- on supprime tous les fils de ce nœud, ce qui supprime une proportion $D^{-\ell_i}$ des nœuds de l'arbre de départ.

Par hypothèse, on supprime une proportion au plus $\sum D^{-\ell_i} \leqslant 1$ des nœuds de l'arbre de départ, c'est-à-dire qu'il reste toujours au moins un nœud tant que l'on n'a pas traité tous les ℓ_i.

▷ 3. On utilise l'inégalité de convexité (si $\sum_i p_i = 1$, et f convexe, alors $\sum_i p_i f(x_i) \geqslant f\left(\sum_i p_i x_i\right)$) avec la fonction $x \mapsto D^x$ qui est convexe. Les x_i sont les $-\log_D p_i - l_i$.

On obtient alors $D^{H_D(p)-L(h)} \leqslant \sum_i p_i D^{-\log_D p_i - l_i} = \sum_i p_i \dfrac{D^{-l_i}}{p_i} \leqslant 1$.

Donc, en prenant le log des deux côtés, on obtient, $H_D(p) - L(h) \leqslant 0$.

Pour la deuxième partie de l'inégalité, on va poser : $\forall i, l'_i = \lceil -\log_D p_i \rceil$. On commence par vérifier que ce sont des candidats possibles pour les longueurs d'un code. En effet, $\sum_i D^{-\lceil -\log_D p_i \rceil} \leqslant \sum_i D^{\log_D p_i} \leqslant 1$.

On a alors $\sum_i p_i l'_i - H_D(p) = \sum_i p_i \left(\lceil -\log_D p_i \rceil + \log_D p_i\right) \leqslant 1$.

Or $L_{inf} \leqslant \sum_i p_i l'_i$, donc

$$L_{inf} \leqslant H_D(p) + 1$$

Exercice 2 L'algorithme de Shannon

On fixe $D = 2$ et on suppose $p_1 \geqslant p_2 \geqslant ... \geqslant p_k$. On pose $q_s = \sum_{i=1}^{s-1} p_i$. La méthode de Shannon consiste à écrire pour chaque i un développement dyadique de q_i :

$$q_i = \frac{a_1(i)}{2^1} + \frac{a_2(i)}{2^2} + \frac{a_3(i)}{2^3} + \cdots$$

et à prendre pour code de x_i : $a_1(i)...a_{l_i}(i)$ avec $l_i = \lceil -\log_2 p_i \rceil$.

En d'autres termes, x_i est l'écriture en base 2 de q_i, tronquée à l_i chiffres.

▷ 1. Donner le code obtenu pour

p_1	p_2	p_3	p_4	p_5	p_6
0,27	0,23	0,2	0,15	0,1	0,05

Calculer L et la comparer à l'entropie.

▷ 2. Montrer que ce code est préfixe.

▷ 3. Montrer que la longueur moyenne L vérifie $H_2(p) \leqslant L \leqslant H_2(p) + 1$.

▷ 4. Ce codage est-il optimal ?

Solution Ce tableau est complété au fur et à mesure des questions :

i	p_i	l_i	q_i	Shannon	Fano	mystère
1	0,27	2	0	00	00	10
2	0,23	3	0,27	010	01	00
3	0,2	3	0,5	100	100	01
4	0,15	3	0,7	101	101	110
5	0,1	4	0,85	1101	110	1110
6	0,05	5	0,95	11110	111	1111
			$L =$	2,93	2,5	2,42

▷ 1. L'entropie est 1,68.

▷ 2. Les écritures des codes sont croissantes, donc si x_i est préfixe de x_j pour $j > i$, alors il est aussi préfixe de x_{i+1}. On va montrer que ce n'est pas le cas. On regarde donc la différence entre q_i et q_{i+1} :
$\lceil -\log(q_{i+1} - q_i) \rceil = \lceil -\log p_i \rceil = l_i$. Donc la différence d'écriture se fait sur le l_i[e]bit. Comme les deux codes sont de longueurs au moins l_i, ils sont distincts.

▷ 3. Évident grâce à la définition des l_i.

▷ 4. On verra un meilleur codage avec l'exercice suivant.

Exercice 3 L'algorithme de Fano

Fixons $D = 2$ et supposons $p_1 \geqslant p_2 \geqslant ... \geqslant p_k$. On regroupe les u premiers objets où u est le plus petit entier tel que

$$p_1 + ... + p_u \geqslant \frac{1}{2}$$

Nous obtenons ainsi une partition M de l'ensemble des objets en deux sous-ensemble M_0 et M_1. Soient π_0 et π_1 les probabilités respectives de ces ensembles. Nous appliquons à M_0 l'algorithme de dichotomie pour obtenir une partition $M_0 = M_{00} + M_{01}$ où $M_{00} = \{x_1, ..., x_v\}$ et v est le plus petit entier tel que $p_1 + ...p_v \leqslant \pi_0/2$. On applique à M_1 ce même algorithme et on obtient la partition $M_1 = M_{10} + M_{11}$. On continue l'algorithme ainsi de suite jusqu'à ce qu'on ne puisse plus appliquer la dichotomie, ce qui se produit quand un ensemble M_a ne comporte plus qu'un élément et a est alors le code de cet élément.

▷ 1. Reprendre les questions 2.1, 2.2 et 2.4.

Solution

▷ 1. $L = 2{,}5$

▷ 2. Le code est préfixe par construction, en effet, les codes sont les feuilles d'un arbre binaire, donc aucun code ne peut être préfixe d'un autre.

▷ 4. On verra un meilleur codage avec l'exercice suivant.

Exercice 4 L'algorithme mystère

▷ 1. Donner un algorithme pour obtenir un codage préfixe optimal. Le faire tourner sur l'exemple de la question 2.1 et comparer la longueur moyenne L obtenue avec l'entropie de la distribution.

Solution

▷ 1. On utilise l'algorithme de Huffman :
- A chaque étape, on prend les deux lettres les moins fréquentes, et on les place comme sœurs dans l'arbre associé au codage.
- On réunit ces deux lettres en une seule de probabilité la somme des deux, qui sera le père des deux.
- On ajoute cette nouvelle lettre aux autres, et on recommence.

Ce codage est optimal. En effet, le nombre d'arbre binaires à k éléments est fini, il existe donc un arbre optimal. Considérons la lettre la moins fréquente, on peut l'échanger avec une feuille de profondeur maximale et diminuant (au sens large) la longueur moyenne de codage. Donc il existe un codage optimal où la lettre la moins fréquente est à la profondeur maximale.

Par un raisonnement analogue, on montre qu'il existe un codage optimal ou la deuxième lettre moins fréquente est sœur de la précédente. On peut alors fusionner ces deux lettre en une nouvelle, de fréquence somme des fréquences des deux lettres, et poursuoivre l'algorithme.